



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/670,350	09/26/2003	Hideharu Mochizuki	030712-10	5253

22204 7590 03/26/2007
NIXON PEABODY, LLP
401 9TH STREET, NW
SUITE 900
WASHINGTON, DC 20004-2128

EXAMINER

BROWN JR, NATHAN H

ART UNIT	PAPER NUMBER
----------	--------------

2121

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/26/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No. 10/670,350	Applicant(s) MOCHIZUKI, HIDEHARU	
	Examiner Nathan H. Brown, Jr.	Art Unit 2121	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE (3) MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 03 January 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-4 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-4 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

Examiner's Detailed Office Action

1. This Office Action is responsive to the communication for application 10/670,350, filed January 3, 2007.
2. Claims 1-4 are pending.
3. After the previous office action, claims 1-4 stood rejected.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 1, 2, and 3 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Cross II et al.* "Control Structure Diagrams for Ada 95", 1996 in view *Hendrix et al.*, "Visual Support for Incremental Abstraction and Refinement in Ada 95", 1998.

Regarding claim 1. (Currently Amended) *Cross II et al.* teach a graphical interface method in an outline-processor for a computer system (*see* Abstract, *Examiner interprets CSD to be a graphical interface method in an outline-processor for the computer system GRASP.*) having input means for entering data, data storage means for storing data, and a display screen means for

Art Unit: 2121

creating, editing and viewing of a program (*see* p. 145, col. 1, “The CSD window, shown in Figure 8, is a full-function text editor with the additional capability to generate, display, edit, and print CSDs. The File and Edit options are similar to traditional text editors.”, *Examiner interprets the functions provided by the CSD window to comprise a means for entering data, a data storage means, and a display screen for creating, editing and viewing a program.*).

Cross II et al. do not teach providing a graphical presentation of a program shown as diagram-display having a plurality of outline-display frames connected by lines forming an inverted tree hierarchical structure. *Hendrix et al.* do teach providing a graphical presentation of a program shown as diagram-display having a plurality of outline-display frames connected by lines forming an inverted tree hierarchical structure (*see* p. 153, Fig. 2, *Examiner interprets the structure of the CSD outline to be an inverted tree hierarchical structure (see Aoyama et al., “Design Specification in Japan: Tree-Structured Charts”, 1989).*).

Cross II et al. do not teach displaying an expanded view on the display screen containing source codes of each respective outline-display frame upon the activation of the expanded view of the respective outline-display frame by an input device so that the source codes of the program is displayed with clarity for viewing, so that the program may be edited, wherein the expanded view of each respective outline-display frame is shown simultaneously on the same display screen as the inverted tree hierarchical structure of the program. However, *Hendrix et al.* do teach displaying an expanded view on the display screen containing source codes of each respective outline-display frame upon the activation of the expanded view of the respective outline-display frame by an input device so that the source codes of the program is displayed

Art Unit: 2121

with clarity for viewing (*see* p. 154, col. 1, “A user can select portions of code according to control structure boundaries, program module boundaries, or arbitrary boundaries, and then fold them into the single CSD symbol shown in Figure 3”, *Examiner interprets “select portions of code” to mean clicking on the CSD folding symbol with a mouse.*), so that the program may be edited (*see* p. 155, col. 2, “Users of GRASP have direct visual support for incremental development and stepwise refinement through folded CSDs. Users can create initially folded CSDs to represent regions of code that need refinement or elaboration. As these regions are incrementally refined, they can be individually folded again to reduce the visual clutter and allow the user to focus on the current region of code being developed.”, *Examiner interprets “incrementally refined” to mean edited.*), wherein the expanded view of each respective outline-display frame is shown simultaneously on the same display screen as the inverted tree hierarchical structure of the program (*see* p. 155, col. 1, Fig. 7).

Regarding claim 2. (Currently Amended) *Cross II et al.* teach a graphical interface method in an outline-processor (*see* above), further comprising displaying an argument frame in the vicinity of a respective frame shown in the displayed program inverted tree hierarchical structure, wherein the argument frame shows the function of the source codes in each respective frame (*see* p. 153, col. 1, Fig. 2, *Examiner notes that function “accept REQUEST(p) (D : DATA)” is shown in an argument frame in the vicinity of the respective frame for “task body TASK_NAME”.*).

Regarding claim 3. (Currently Amended) over *Cross II et al.* teach a graphical interface method in an outline-processor (*see* above), further comprising displaying frame lines of each outline-display frame in the program inverted tree hierarchical structure with a first thickness

Art Unit: 2121

(see p. 145, Fig. 6, Examiner interprets the "CSD Unit Symbol" for "package specification" to have a first thickness.); and displaying frame lines of the expanded view of each respective outline-display frame with a second thickness that is thicker than the first thickness (see p. 145, Fig. 5, Examiner interprets the "CSD Box Notation" for "package specification" to have a second thickness that is thicker than the first thickness.).

It would have been obvious at the time the invention was made to persons having ordinary skill in the art to combine *Cross II et al.* with *Hendrix et al.* to improve the comprehension efficiency of software and, as a result, improve reliability and reduce costs during design, implementation, testing, maintenance and reengineering.

6. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over *Cross II et al.* in view of *Hendrix et al.*, "Providing Enhanced Visual Support for Software Development and Maintenance", 1998.

Regarding claim 4. (Previously Presented) *Cross II et al.* teach a graphical interface method in an outline-processor (see above). *Cross II et al.* do not teach displaying a most recently activated expanded view of the respective outline-display frame on top of other expanded views of outline-display frames. *of Hendrix et al.* do teach displaying a most recently activated expanded view of the respective outline-display frame on top of other expanded views of outline-display frames (see p. 25, col. 2, Fig. 6, Examiner interprets Fig. 6 to show an expansion of

conversion_test into base_conversion.abd and conversion_test.abd, where conversion_test.abd is the most recently activated expansion and overlaps base_conversion.abd.).

It would have been obvious at the time the invention was made to persons having ordinary skill in the art to combine *Cross II et al.* with *Hendrix et al.* to provide automatic visualization of software control structure and complexity to support development, maintenance, reverse engineering, and reengineering.

Response to Arguments

7. Applicant's arguments filed January 3, 2007 have been fully considered but they are not persuasive.

Rejection of Claims 1-3 Under 35 U.S.C. §103(a)

Applicant argues:

In the rejections of claims 1-4 over Cross II and Hendrix, the Examiner interprets Control Structure Diagrams (CSD's) of Hendrix to be equivalent to Applicant's "outline-display frames" and the structure of the CSD as shown in Fig. 2 of Hendrix as equivalent to Applicant's "outline-display frames connected by lines forming an inverted tree hierarchical structure".

Examiner responds:

Both applicant's outline-display frames and Hendrix's CSD's show a hierarchical decomposition of some body of code. In this respect, Hendrix's CSD's are equivalent to applicant's outline-display frames. It should also be noted that the hierarchical decomposition is a form acyclic directed graph where the root node is at the highest level.

Art Unit: 2121

Applicant argues:

The Examiner then interpreted Hendrix as teaching displaying an expanded view on the display screen containing source codes of each respective outline-display frame upon the activation of the expanded view of the respective outline-display frame by an input device so that the source codes of the program is displayed with clarity for viewing. Applicant respectfully submits that neither Cross II nor Hendrix teaches, discloses or suggests the outline-display frames of the claimed invention. In the CSD of Cross II and Hendrix, a functional block of source code is surrounded by a box denoting the function of the box. The source code can then be viewed either in code form or in functional block form. However, in Applicant's claimed invention, the outline-display frames show a main program, subroutines, and their relationship (see Specification, P. 7, In. 11 to p. 8, In. 14).

Examiner responds:

The CSD of Cross II and Hendrix show a main program, subroutines, and their relationship (*see* Hendrix et al., "Visual Support for Incremental Abstraction and Refinement in Ada 95", p. 155, Fig. 7). The source code can then be viewed either in code form or in functional block form in the CSD of Cross II and Hendrix as an unfolded CSD symbol or as a folded CSD symbol, respectively.

Applicant argues:

Additionally, Applicant respectfully submits that neither Cross II nor Hendrix teaches, discloses or suggests the inverted tree structure of the claimed invention. As noted in previous responses, Applicant repeats the distinguishing feature of Applicant's claimed inverted tree structure compared with the structure of Cross II and Hendrix. Specifically, Cross II and Hendrix teach multi-level or nested loops identified by vertical lines identifying each nested level of nested loops and a CSD unit symbol identifying a routine or a functional module (see Fig. 2), where the source codes with nested loops shown in Hendrix are made easier to read by using CSD's to label and show structural and control information for each module and to allow control structures, such as loops, to be selectively displayed by hiding or folding portions of nested loops, thereby improving the readability of a long program. Hendrix's Fig. 2 merely shows multi-level or nested loops identified by vertical lines identifying each nested level of nested loops and a CSD unit symbol identifying a routine or a functional module. That is, the source codes with nested loops shown in Hendrix are made easier to read by using CSD's to label and show structural and control information for each module and to allow control structures, such as loops, to be selectively displayed by hiding or folding portions of nested loops thereby

Art Unit: 2121

improving the readability of a long program. By contrast, in Applicant's claimed invention, each outline-display frame can be expanded to show the source code in an expanded view. The structure of the multiple outline-display frames and multiple expanded views is an inverted tree structure, having multiple branches and many leaves upon each branch. Because the relationship of each subroutine to the main program is shown diagrammatically in Applicant's claimed invention, (see Fig. 1), each branch may have multiple sub-branches, similar to a tree in structure.

Examiner responds:

There is no contrast between Applicants claimed invention and the CSDs of Cross II and Hendrix, except in perhaps in the way the diagrams look, which is irrelevant to the real-world result. In comparison, each outline-display frame in the Applicant's claimed invention can be expanded to show the source code in an expanded view as can each CSD symbol can be unfolded to show source code in an expanded view (*see* Hendrix et al., p. 155, Figs. 7-8). Examiner notes that Hendrix et al. do not show functional nesting below one level, but code nesting is shown down to four levels (*see* Hendrix et al., "Visual Support for Incremental Abstraction and Refinement in Ada 95", p. 153, Fig. 2)

Applicant argues:

Further, Hendrix and Cross II only show that source codes or blocks of source codes are displayed or listed in a sequential manner. That is, Hendrix does not teach, disclose or suggest an inverted tree hierarchical diagram of a program such as shown in Applicant's Fig. 1 or recited in claim 1 of the present invention. For example, Hendrix method does not and cannot show block or frame 105 connected to a higher level block or frame 101 as shown in Fig. 1 of Applicant's disclosure. Likewise, Hendrix does not and cannot show block or frame 103 connected to block or frame 101.

Examiner responds:

Hendrix shows multiple folded CSDs connected to a higher level frame (*see* Hendrix et al., "Visual Support for Incremental Abstraction and Refinement in Ada 95", p. 155, Fig. 8). See Appendix to Office Action.

Art Unit: 2121

Applicant argues:

Finally, Applicant respectfully submits that neither Cross II nor Hendrix teaches, discloses or suggests showing the expanded view of each respective outline-display frame simultaneously on the same display screen as the inverted tree hierarchical structure of the program, as is recited in claim 1. In Hendrix, a user can optionally select to view a "landscape" view alongside the source code view (see p. 155-156). The highly compressed view of the CSD and/or source code is shown in a companion window displayed alongside the ordinary CSD and/or source code window. This is different than the claimed invention of claim 1, where the outline-display frames are shown with expanded views simultaneously on the same display screen.

Examiner responds:

Presumably, the user can optionally select *not* to view a "landscape" view alongside the source code view and thus view the folded CSD symbols (outline-display frames) along with the expanded views simultaneously on the same display screen as in Hendrix et al., p. 155, Fig. 7.

Applicant argues:

Because Cross II and Hendrix, either alone or in combination, do not teach or suggest each and every element of independent claim 1, Applicant respectfully submits that Cross II and Hendrix do not render claim 1 unpatentable. The arguments set forth above with respect to claim 1 are also applicable to the rejection of claims 2-4 over Cross II and Hendrix.

Examiner responds:

Cross II and Hendrix in combination teach each and every real-world result of independent claim 1 and renders claim 1 and 2-4 unpatentable. Examiner asserts that Cross II and Hendrix's CSDs do not have to look exactly like Applicant's outline-display frames to have the same functional result.

Applicant argues:

Additionally, with respect to claim 2, neither Hendrix nor Cross II teach, disclose or suggest an argument frame displayed in vicinity of a respective frame in a displayed program inverted tree hierarchical structure, as amended in claim 2. Applicant respectfully asserts that Fig. 2 of Cross does not show any box/block/frame, if the module of source code in Fig. 2 is considered as equivalent to Applicant's frame.

Examiner responds:

Cross does show a frame around the text for "Function in_list (...)" which examiner interprets to be equivalent to item 108 in Applicant's Fig. 1. That fact that Cross' frame contains the return value for the function does not make Cross' frame non-equivalent to applicant's as both do contain the function signature.

Applicant argues:

With respect to the rejection of claim 3, the Examiner alleges that Fig. 3 of Hendrix teaches changing thickness of frame line before and after expansion. However, as previously Fig. 3 of Hendrix merely shows an example of a CSD folding symbol with thick and thin lines drawn in the same symbol block, and there is no relation between Fig. 3 and Fig. 6 showing the changing in thickness of any lines when the block is expanded. Further, the purpose of showing different line thicknesses in the CSD folding symbol in Fig. 3 appears to have no relation to improving the understandability of the folding symbol.

Examiner responds:

The relation between Fig. 3 and Fig. 6 shows that unfolding the CSD symbol (with thick lines) generates a frame (with thin lines) around the function text. This is directly shown in Figs 7 and 8 on p. 155.

Applicant argues:

With respect to the rejection of claim 4, the Examiner alleges that Fig. 6 of Hendrix teaches displaying a most recently activated expanded view of the respective outline-display frame on top of other expanded views of outline-display frames. However, Fig. 6 of Hendrix merely shows an example of a folding CSD structure, where parts of the CSD

Art Unit: 2121

are folded and parts are expanded. By contrast, in applicant's claimed invention, multiple expanded views can be opened on a single display screen, with the most recently activated expanded view shown on top of the less recently activated expanded views (see Fig. 2). The requirements for establishing a *prima facie* case of obviousness, as detailed in first, there must be some MPEP § 2143 - 2143.03 (pages 2100-122 - 2100-136), are: suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference to combine the teachings; second, there must be a reasonable expectation of success; and, finally, the prior art reference (or references when combined) must teach or suggest all of the claim limitations.

Examiner responds:

Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cross II et al. in view Beaudouin-Lafon, "Novel Interaction Techniques for Overlapping Windows", 2001:

Regarding claim 4. Cross H et al. teach the outline-processor according to claim 1 (see above). Cross H et al. do not teach the outline-processor according to claim 1, wherein when the plurality of programs is outline-displayed ... in a plural number, a newest outline-display ... is displayed in front. Beaudouin-Lafon does teach an outline-processor according to claim 1, wherein when the plurality of programs is outline-displayed ... in a plural number, a newest outline-display ... is displayed in front (see p. 153, §TABBED WINDOWS, Fig. 2, "Leafing facilitates this look-up phase: in addition to popping up the tabs while mousing over them, the corresponding pages are displayed on top." Examiner interprets the tabbed window to be an outline-display of a plurality of programs (one tab per program) and the page corresponding to a popped up tab to contain the source code for a program.).

Applicant argues:

As Cross II and Hendrix are deficient as discussed above, their combination in the pending §103(a) rejections is improper. Because Cross II and Hendrix, either alone or in combination, do not teach or suggest each and every element of claims 2-4, Applicant respectfully submits that Cross II and Hendrix do not render claims 2-4 unpatentable.

Examiner responds:

Examiner disagrees. Cross II et al. and Hendrix in combination teach or suggest each and every element of claims 2-3 while Cross II et al. in combination with Beaudouin-Lafon teach every element of claim 4.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).


A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nathan H. Brown, Jr. whose telephone number is 571-272- 8632. The

Art Unit: 2121

examiner can normally be reached on M-F 0830-1700. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Anthony Knight can be reached on 571-272-3687. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

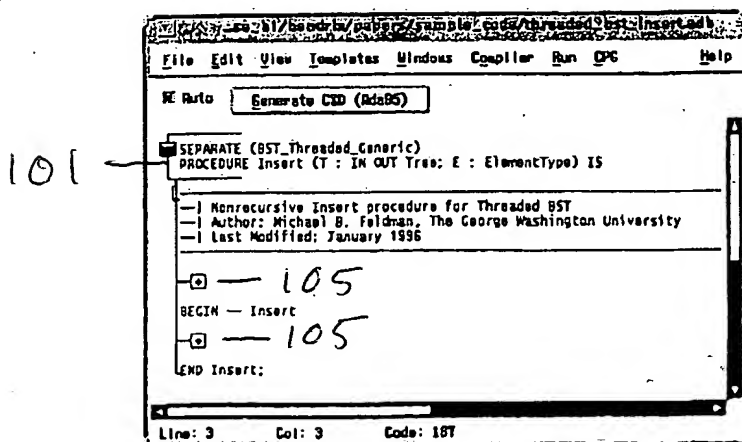


Anthony Knight
Supervisory Patent Examiner
Tech Center 2100

Nathan H. Brown, Jr.
March 19, 2007

Art Unit: 2121

Appendix To Office Action



BEST AVAILABLE COPY